

My Very Own CMS

h0uz3

Neue Ansätze für ein Content Management System

Übersicht

Part 1: The Rant

Part 2: Wunschliste für ein tolles CMS

Part 3: Call for Participation

Part 1: The Rant

Dokumente haben nichts in einer relationalen Datenbank verloren!

- Inhalte sind Dokumente, lasst Gnade walten!
- „Dokumente in eine relationale DB zu stopfen ist, als würde man Club-Mate entkoffeinieren.“
- Übernahme in andere Systeme geht nur mit speziellem Converter
- Formatieren der Dokumente durch Auslesen mehrerer Tabellen und zusammenfrickeln der Inhalte

Grottiges Template-Handling

- Templates beschreiben Seite
- Struktur der Ausgabe erfolgt aber über Einstellungen im CMS („Bereiche“)
- Templates sollten den kompletten Seitenaufbau beschreiben
- Templates sind umständlich und Syntax ist CMS-spezifisch

Ein CMS muss nicht DAU-Proof sein

- Klicki-Bunti-Oberfläche ist bäh
- Tun so, als könnte es jeder einrichten
- Aber nicht jeder sollte ein CMS einrichten!
- Unnötiges Login-System mit Benutzerverwaltung

Part 2: Wunschliste für ein CMS

Warum noch ein CMS?

- Bestehende CMS sind alle doof
- „Sowas wollte ich schon immer machen“
- Verfolgen neuer Ansätze (alsob!)

CMS für den Nerd

- Dokumente hochladen per 'scp'
- Kein doofer Webeditor
- Man kann emacs, joe oder vi nehmen (oder sed)
- Anwendungen können publishen

Simplifizierung

- Einlesen von XML-Dokumenten
- Anreichern um Navigations-Informationen
- Formatieren mittels XSL-T
- Ausliefern an Client (kann ein Browser sein)

Datenhaltung

- Keine relationale Datenbank sondern dokumentenorientierte Datenbank
- Knallhart: XML-Dokumente im Dateisystem
- Performance-Trick: Alle Dokumente im RAM halten
- (RAM ist billig aber schnell!)

Beiträge sind einfache XML-Dokumente

```
<content>
  <title>Mein Content</title>
  <author>h0uz3</author>
  <brief>Einleitung / Kurzbeschreibung</brief>
  <body>
    <p>Absatz 1</p>
    <p>Absatz 2, <e>der ist wichtig</e></p>
  </body>
</content>
```


Templates sind in XSL verfasst

- Es gibt etliche Tools, die XML per XSL formatieren
→ Einfache Entwicklung von Templates
- Maximale Kontrolle über Ausgabe-Format
- HTML-Features nach Lust und Laune verwendbar
- Das ist ultimativ flexibel!!!!111einselbhundertelf

Tagging

- Tags statt Rubriken!
- Dokumente werden getaggt und sind unter diesen Tags zu finden
- Navigation hangelt sich an Tag-Baum entlang
- Beispiel: fotos / veranstaltung / tanturnier

Bilder, Videos und Sound

- Einbinden von flickr, Vimeo und anderen
 - Direkt durch CMS
 - Per Template
- Multimedia-Daten selbst veröffentlichen
 - Eingebauter Player und Slideshow

Plugin-System

- Modulare Plugins bereiten die Daten auf (Formatierung, Lesen aus Datenbank)
- Plugins reagieren auf Events (Datei-Upload)
- Beispiel: Atom-Feed
 - Liest aktuellste Postings aus Datenstruktur
 - Gibt XML aus (ach, echt?)

Publishing

- Inhalte werden als ZIP-Datei eingeliefert
 - XML-Dokument enthält Text
 - HREFs auf evtl. enthaltene Bilder, Videos, Sounds
- Upload per Webservice, scp oder WebDAV
- Veröffentlichen von Shell-Script aus möglich

Präsentation

- Navigation anhand der Dokument-IDs
 - z.B. Mein-Urlaub-an-der-Nordsee
- Navigation anhand von Tags statt Rubriken
 - urlaub, freizeit, nordsee
- SEF-URLs:
 - <http://saschakaupp.com/freizeit/urlaub/nordsee/Mein-Urlaub-an-der-Nordsee.html>
 - Tags sind dabei optional

Services

- Kommentarfunktion
- RSS/Atom-Feeds
- Sehr stark vereinfachtes Backup
 - Dateien sichern und fertig!

Part 3: Call for Participation

Jeder darf mitmachen

- h0uz3@foto23.com
- Mailingliste wird eingerichtet
- Wird ein Repository geben (vermutlich git)
- Es dürfen auch blutige Anfänger mitmachen

One more thing...

Implementierung

- Minimale Java Servlet/JSP-Infrastruktur
- Verzicht auf sonstige Frameworks
- Controller-Servlet für Ablauf
- Daten speichern als einfache Struktur im Dateisystem

Programmier-Anforderungen

- Java 6
- Sprachfeatures bestimmungsgemäß nutzen
- Core soll weniger als 1000 LOC haben
- Übersichtliche, kleine Klassen mit klarer Vererbungslinie

Danke für's zuhören!

- Bei Interesse:
 - h0uz3@foto23.com